EXPERIMENT NO. 1

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

LED Blinking using software delay.

OBJECTIVES:

- To understand LPC2148 microcontroller & its various ports and registers.
- To interface LED with LPC2148.
- To program LPC2148 using software method to generate delay.

CO and PO MAPPED :

CO –

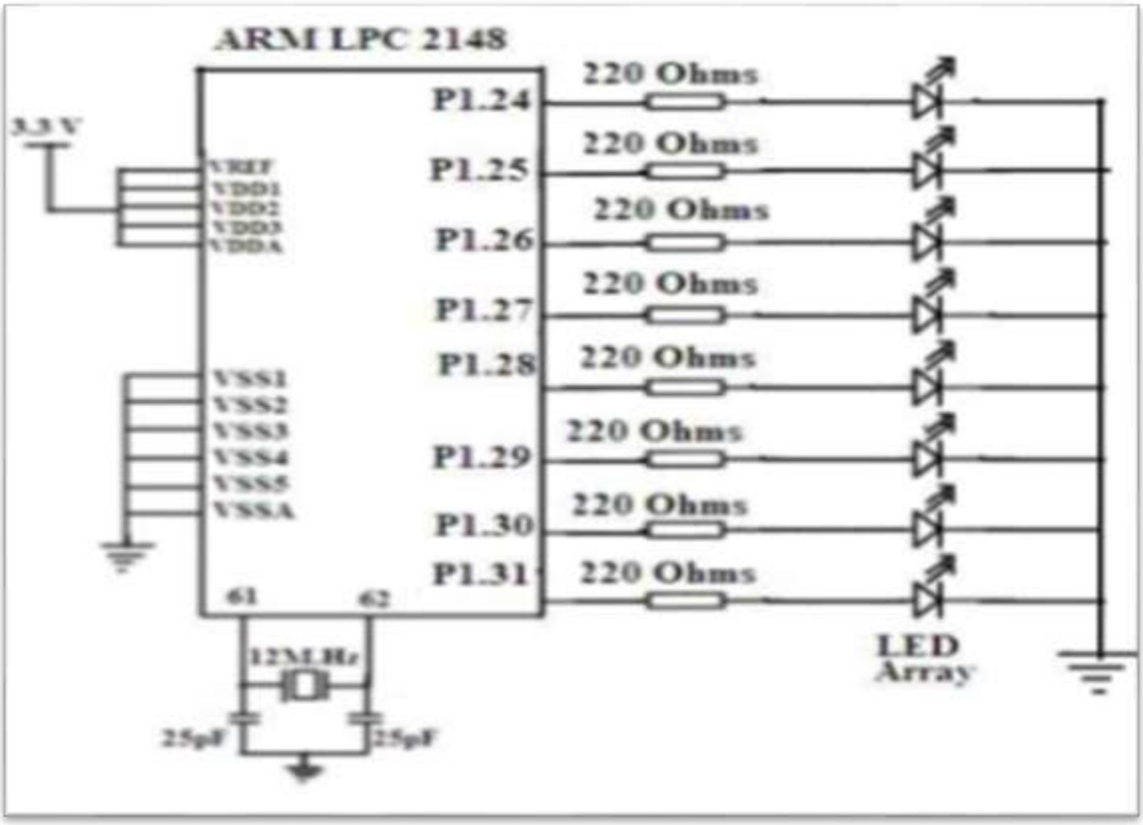
PO –

REQUIREMENT: LPC2148 board, 12V adaptor, USB cable

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Observe the LED blinking.

INTERFACING DIAGRAM:



THEORY:

LED stands for Light Emitting Diode. Light emitting diodes are popularly used display components used to indicate the ON and OFF state of system. These are also used to realize various counters like binary counters experimentally. These LEDs can be easily interfaced with the port pins of any microcontroller by using current limiting resistors. To find the value of this resistor, we use the forward voltage and forward current.

Algorithm:

1. Initialize the ports of LPC2148 using PINSEL2 and IODIR1 registers.
2. Write delay function using software method.
3. Make the port lines high using IOSET1 register to glow LED.
4. Call delay.
5. Make the port lines low using IOCLR1 register to make LED off.
6. Call delay.
7. Go to step 2.

CONCLUSION:

QUESTIONS:

1) List and explain the functions of GPIO registers of LPC2148.

2) Explain pin connect block of ARM.

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

EXPERIMENT NO. 2

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

LED Blinking using TIMER.

OBJECTIVES:

- To understand LPC2148 microcontroller & its various ports and registers.
- To interface LED with LPC2148.
- To program LPC2148 using timer to generate delay.

CO and PO MAPPED :

CO –

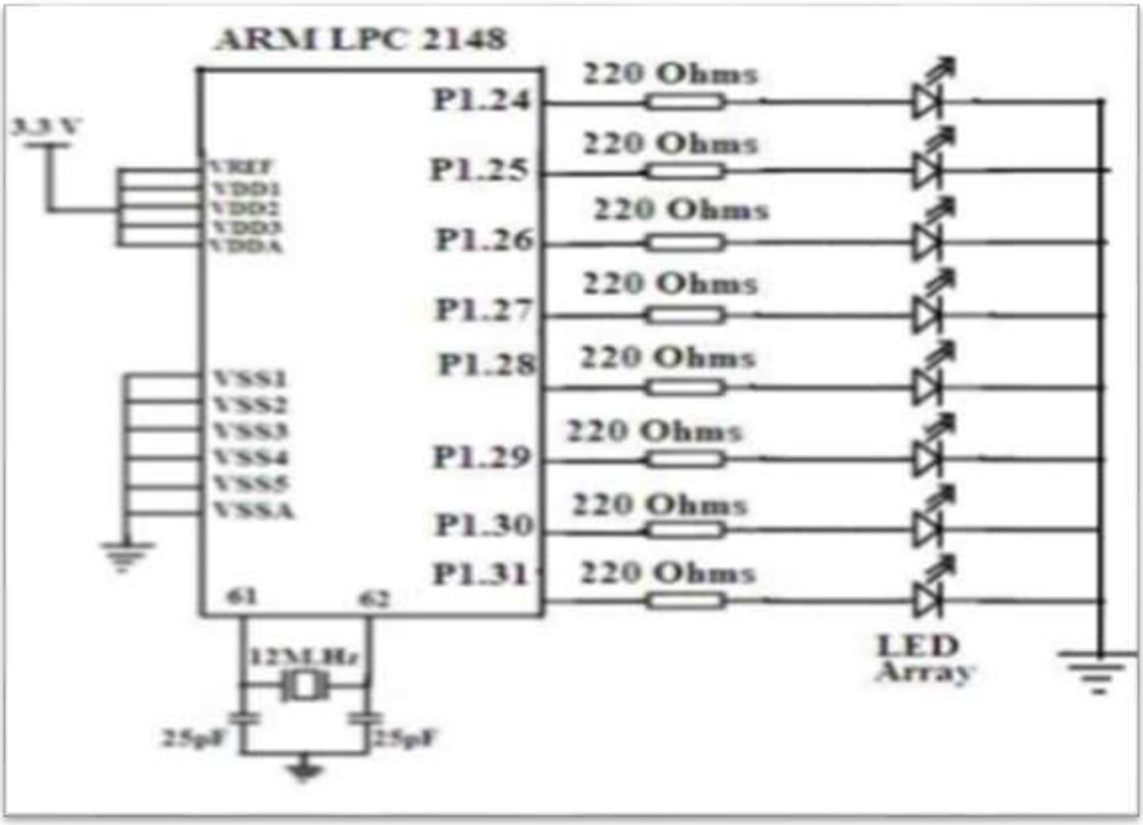
PO –

REQUIREMENT: LPC2148 board, 12v adaptor, USB cable

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Observe the LED blinking.

INTERFACING DIAGRAM:



THEORY:

LPC2148 Timer/Counter:

- LPC2148 has two 32-bit timers/counters: Timer0/Counter0 & Timer1/Counter1.
- The Timer/Counter is designed to count cycles of the peripheral clock (PCLK) or an externally supplied clock, and can optionally generate interrupts or perform other actions at specified timer values, based on four match registers.
- It counts the clock from either of these clock sources for its operation.
- LPC2148 Timer/Counter can generate an interrupt signal at specified time value.
- LPC2148 has match registers that contain count value which is continuously compared with the value of the Timer register.
- When the value in the Timer register matches the value in the match register, specific action (timer reset, or timer stop, or generate an interrupt) is taken.
- Also, LPC2148 has capture registers which can be used to capture the timer value on a specific external event on capture pins.
- Registers Used in LPC2148 Timer:

Register	Description
IR	Interrupt Register: The IR can be read to identify which of 6(4-match, 2-Capture) possible interrupt sources are pending. Writing Logic-1 will clear the corresponding interrupt.
TCR	Timer Control Register: The TCR is used to control the Timer Counter functions(enable/disable/reset).
TC	Timer Counter: The 32-bit TC is incremented every PR+1 cycles of PCLK. The TC is controlled through the TCR.
PR	Prescaler Register: This is used to specify the Prescaler value for incrementing the TC.
PC	Prescale Counter: The 32-bit PC is a counter which is incremented to the value stored in PR. When the value in PR is reached, the TC is incremented.
CTCR	Count Control Register. The CTCR selects between Timer and Counter mode, and in Counter mode selects the signal and edge(s) for counting.
MR0-MR3	Match Registers: The Match register values are continuously compared to the Timer Counter value. When the two values are equal, actions can be triggered automatically. The action possibilities are to generate an interrupt, reset the Timer Counter, or stop the timer. Actions are controlled by the settings in the MCR register.
MCR	Match Control Register: The MCR is used to control the resetting of TC and generating of interrupt whenever a Match occurs.
CR0 - CR4	Capture Registers. TC value is loaded to this Capture Register when there is an event on the CAPn.0 - CAPn.4
CCR	Capture Control Register. The CCR controls which edges of the capture inputs are used to load the Capture Registers and whether or not an interrupt is generated when a capture takes place.

Configuring Timer :

Use the following sequence for Setting up Timers:

1. Set appropriate value in TxCTCR
2. Define the Prescale value in TxPR
3. Set Value(s) in Match Register(s) if required
4. Set appropriate value in TxMCR if using Match registers / Interrupts
5. Reset Timer – Which resets PR and TC
6. Set TxTCR to 0x01 to Enable the Timer when required
7. Reset TxTCR to 0x00 to Disable the Timer when required

Prescale Register Value Calculation:

- The delay or time required for 1 clock cycle at ‘X’ MHz is given by :
  Delay = 1/(X*1000000) Seconds
- Hence in our case when PR=0 i.e TC increments at every PCLK the delay required for TC to increment by 1 is:
  Delay = (0+1)/(60*1000000) Seconds
- Similarly, when we set PR = 59999 the delay will be:
  Delay = (599999+1)/(60*1000000) Seconds
  Delay = 1mS
  … which boils down to 1/1000 = 0.001 Seconds which is nothing but 1 Milli-Second i.e mS. Hence the delay required for TC to increment by 1 will be 1mS.

Algorithm:

1. Initialize the ports of LPC2148 using PINSEL2 and IODIR1 registers.
2. Write delay function using timer.
3. Make the port lines high using IOSET1 register to glow LED.
4. Call delay.
5. Make the port lines low using IOCLR1 register to make LED off.
6. Call delay.
7. Go to step 2.

CONCLUSION:

QUESTIONS:

1) List and explain the PLL Configuration Registers of LPC2148.

2) Explain the following timer registers of LPC2148:

   i. Prescale Counter Register
   ii. Timer Counter Register
   iii. Match Register
   iv. Match Control Register

3) Explain the steps to generate the delay of 500ms using timer when PCLK = 15Mhz.

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

EXPERIMENT NO. 3

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Interfacing with 16x2 LCD.

OBJECTIVES:

- To understand LPC2148 microcontroller & its various ports and registers.
- To interface 16x2 LCD with LPC2148.
- To program LPC2148 for implementing desired application.

CO and PO MAPPED :

CO –

PO –

REQUIREMENT: LPC2148 board, 12v adaptor, USB cable

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Observe o/p display on 16x2 LCD.

INTERFACING DIAGRAM:

Fig 1 shows the interfacing 16x2 LCD with LPC2148. LCD Data pins D0-D7 are connected to pins P1.16-P1.23 of Port 1. LCD control pins RS, RW and EN are connected to P0.11, P0.20 and P0.10 respectively.

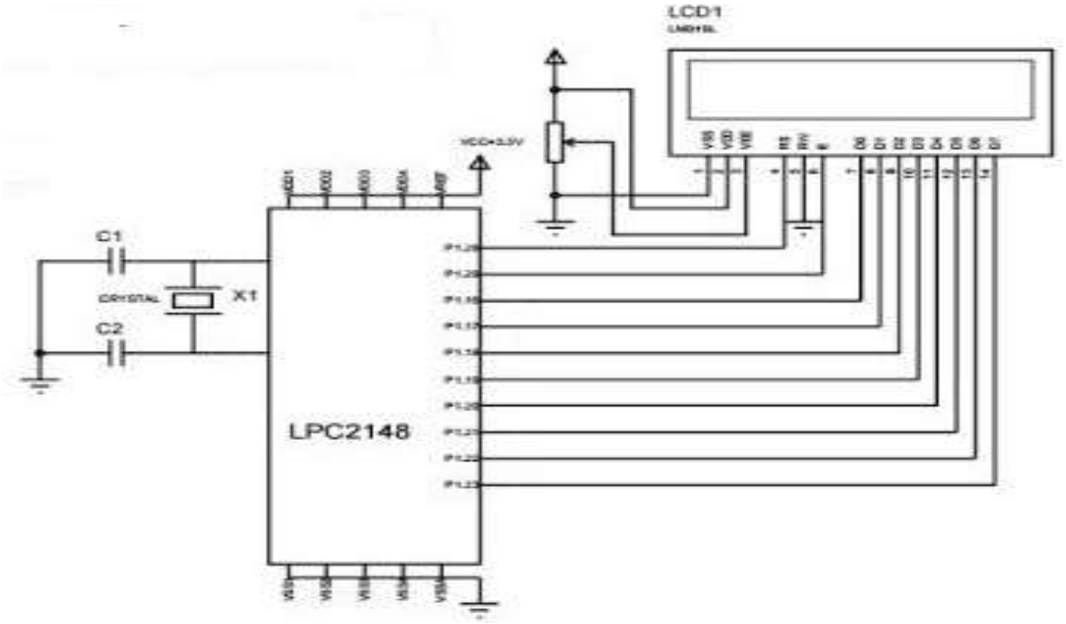


Fig 1: Interfacing 16x2 LCD With LPC2148

THEORY:

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.

LCD 16x2 is 16 pin device which has 8 data pins (D0-D7) and 3 control pins (RS, RW, EN). The remaining 5 pins are for supply and backlight for the LCD. The control pins help us configure the LCD in command mode or data mode. They also help configure read mode or write mode and also when to read or write. LCD 16x2 can be used in 4-bit mode or 8-bit mode depending on the requirement of the application. In order to use it we need to send certain commands to the LCD in command mode and once the LCD is configured according to our need, we can send the required data in data mode to display on it.

Pin Diagram:

16x2 LCD has total 16 pins which can be divided into three categories:

1. Power Control pins: 1,2,3,15,16
2. Control Pins: 4,5,6
3. Data Pins: 7 to 14.

Fig. 2 shows 16x2 LCD pin diagram and functions of every pin. Fig 3 shows the 5x8 pixel character display.

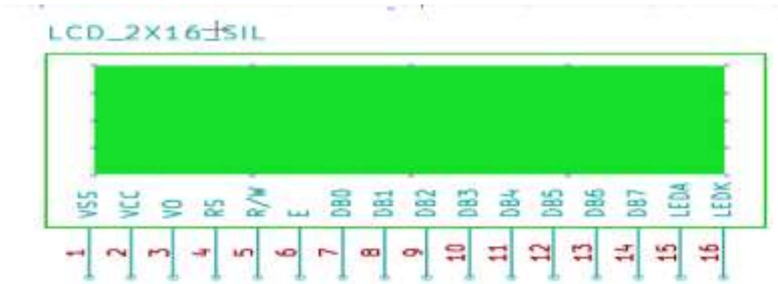


Fig 2 : 16x2 LCD Pin Diagram

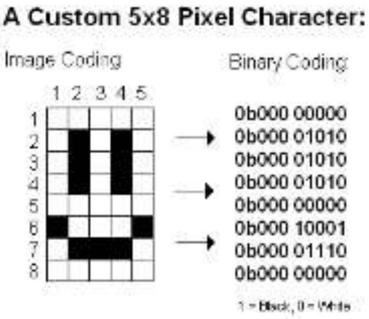
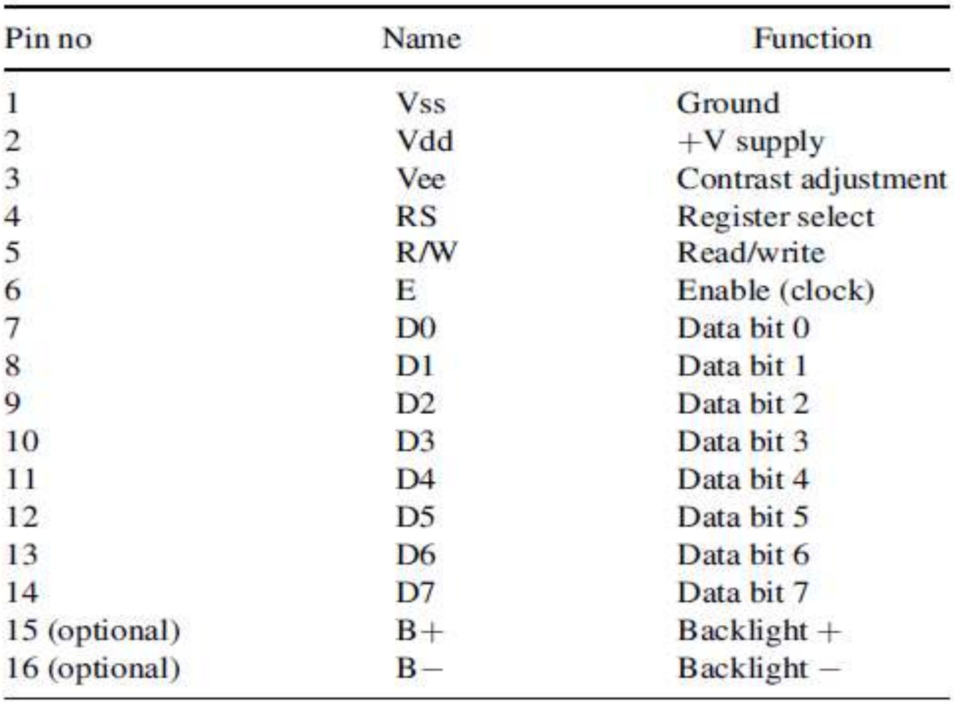


Fig 3 : 5x8 Pixel Character



Pin no	Name	Function
1	Vss	Ground
2	Vdd	+V supply
3	Vee	Contrast adjustment
4	RS	Register select
5	R/W	Read/write
6	E	Enable (clock)
7	D0	Data bit 0
8	D1	Data bit 1
9	D2	Data bit 2
10	D3	Data bit 3
11	D4	Data bit 4
12	D5	Data bit 5
13	D6	Data bit 6
14	D7	Data bit 7
15 (optional)	B+	Backlight +
16 (optional)	B−	Backlight −

The LCD has two registers for its operation :

1. **Command Register:**

- Stores the command instructions given to the LCD.
- A command is an instruction given to LCD to do a predefined task like initializing it,clearing its screen, setting the cursor position, controlling display, etc.

2. **Data Register:**

- Stores the data to be displayed on the LCD.
- The data is the ASCII value of the character to be displayed on the LCD

LCD Control Pin Function :

- **Register Select (RS):**

  1. If RS =0; data present on the data bus will be saved in command register. This Command is 8 bit data which contains information about mode of operation of theLCD Such as cursor position, decrement or increment of the cursor ,clear display, cursor at home, etc.

  2. If RS =1; data present on the data bus will be saved in data register, allowing the user to send data to be displayed on the LCD.Read/Write Select

- **Read/Write pin (RW):**

  The LCD can be operated to perform Read or Write operation.

  1. If R/W=0 ; LCD will perform write operation.

  2. If R/W=1; LCD will operate inread mode.

- **Enable pin (EN):**

  Logic 1 is applied on this pin for few nano second so that the LCD activates and stores the value on data lines into the selected register to perform its normal operation

Data pins:

1. There are total eight data pins named from D0 to D7.

2. Same eight data pins are used to send both, the command and the data.

3. LCD is alpha numeric type and understands only ASCII Code .

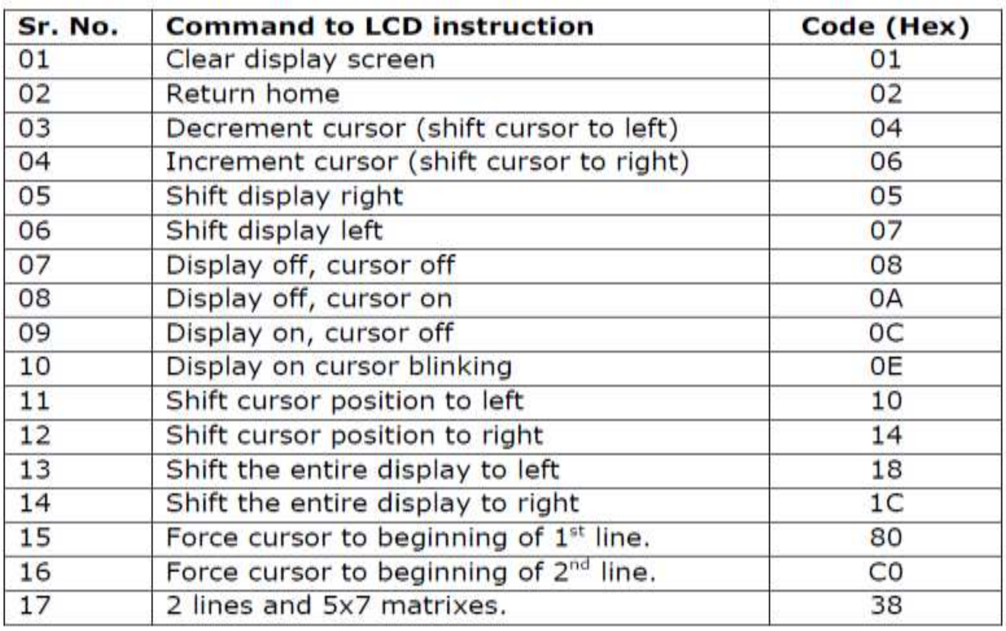
4. In8-bit mode, all eight data pins are connected with the microcontroller whereas in 4-bit mode only four data pins are connected to the microcontroller.

LCD can operate in two modes.

1. **Command Mode:** In this mode, RS pin is at logic 0 and the data on pin 7-14 represents the command which specifies the actions to be performed like clear LCD, go to home position, blink cursor etc.

2. **Data Mode:** In this mode, RS pin is at logic 1 and data op pin 7-14 represents the data to be displayed on LCD.

LCD Command Codes:



Sr. No.	Command to LCD instruction	Code (Hex)
01	Clear display screen	01
02	Return home	02
03	Decrement cursor (shift cursor to left)	04
04	Increment cursor (shift cursor to right)	06
05	Shift display right	05
06	Shift display left	07
07	Display off, cursor off	08
08	Display off, cursor on	0A
09	Display on, cursor off	0C
10	Display on cursor blinking	0E
11	Shift cursor position to left	10
12	Shift cursor position to right	14
13	Shift the entire display to left	18
14	Shift the entire display to right	1C
15	Force cursor to beginning of 1st line.	80
16	Force cursor to beginning of 2nd line.	C0
17	2 lines and 5x7 matrixes.	38

Algorithm for LCD interfacing :

Step 1: Write command subroutine for executing LCD, data subroutine for displaying LCD on LCD and delay subroutine so that the LCD can be given sufficient time to execute command or display data.

Step 2: Command subroutine for LCD should have the following steps

a. Port pins =command in hex code

b. Port pin connected to RS of LCD=0

c. Port pin connected to RW of LCD=0

d. Port pin connected to EN of LCD=1

e. Delay

f. Port pin connected to EN of LCD=0

Step 3: Data subroutine for LCD should have the following steps

a. Port pins =data to be displayed

b. Port pin connected to RS of LCD=1

c. Port pin connected to RW of LCD=0

d. Port pin connected to EN of LCD=1

e. Delay

f. Port pin connected to EN of LCD=0

Step 4: Delay subroutine should store values in the registers and decrement them to generate the desired delay.

Step 5: Write main program to initialize LCD by giving different commands which are Executed by command subroutine, give data which is displayed with the data subroutine.

Step 6: Stop.

CONCLUSION:

QUESTIONS:

1) List the difference between 8 bit interfacing mode and 4 bit interfacing mode and state their advantages and disadvantages.

2) Select correct option:

i) In 4-bit mode of LCD interfacing

a. Lower nibble is sent first then higher nibble.

b. Higher nibble is sent first then lower nibble.

c. Any nibble can be sent at any time.

d. Any 4-bit nibble can be sent at only once.

ii) In 16x2 LCD, which of the following command is used to move the cursor to the last line last column?

a. 0x89

b. 0xC0

c. 0x8F

d. 0xCF

3) Write a function list to initialize the LCD

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

EXPERIMENT NO. 4

Department: E&TC **Academic Year: 2023-24** **Semester: I**

Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Interfacing LPC2148 with GLCD to display image on it.

OBJECTIVES:

- To understand LPC2148 microcontroller & its various ports and registers.
- To interface 128x64 GLCD with LPC2148.
- To program LPC2148 for implementing desired application.

CO and PO MAPPED :

CO –

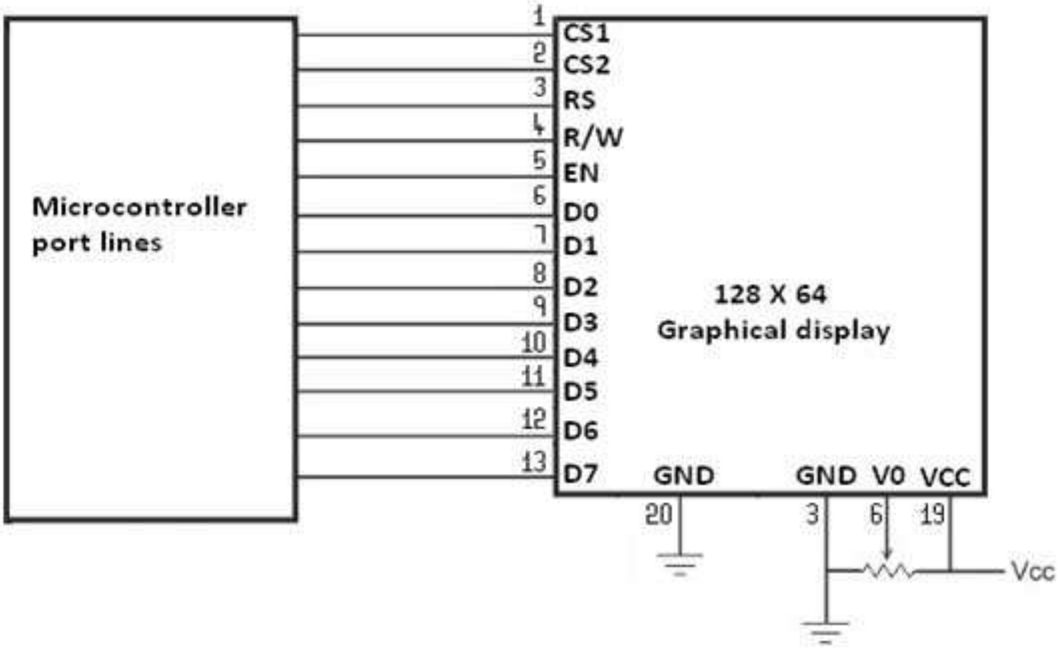
PO –

REQUIREMENT: LPC2148 board, 12v adaptor, 10 pin FRC, USB cable, GLCD adapter

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Connect the 20 pin FRC of GLCD adapter to the CN18 port onARM7 board.
- Observe o/p display on GLCD.

INTERFACING DIAGRAM:



THEORY:

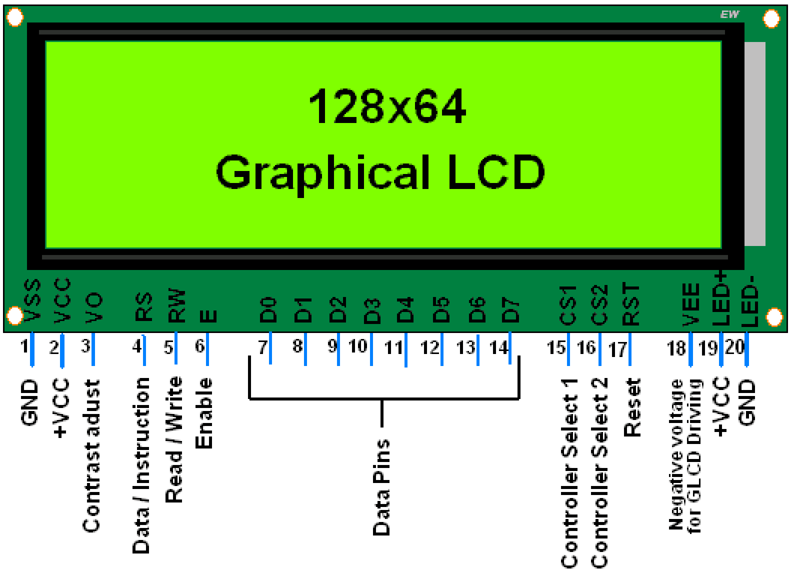
GLCD stands for Graphical Liquid Crystal Display used to display graphics, numbers, strings and special characters as per the application. It is one of the widely used modules in embedded system. It has total 20 pins, connections of which are given in next table:

Graphic LCD extension Header (CN18)

Graphical LCD		**Pin Details**	
	LPC2148	**Connector CN18**	**Description**
D0	P1.16	CN18.11	GLCD Data Bit 0
D1	P1.17	CN18.12	GLCD Data Bit 1
D2	P1.18	CN18.13	GLCD Data Bit 2
D3	P1.19	CN18.14	GLCD Data Bit 3
D4	P1.20	CN18.18	GLCD Data Bit 4
D5	P1.21	CN18.17	GLCD Data Bit 5
D6	P1.22	CN18.16	GLCD Data Bit 6
D7	P1.23	CN18.15	GLCD Data Bit 7
RESET	P1.24	CN18.6	Reset
D/I	P1.25	CN18.5	Data / Instruction Signal
CS1	P1.26	CN18.8	GLCD Chip Select 1
CS2	P1.27	CN18.7	GLCD Chip Select 2

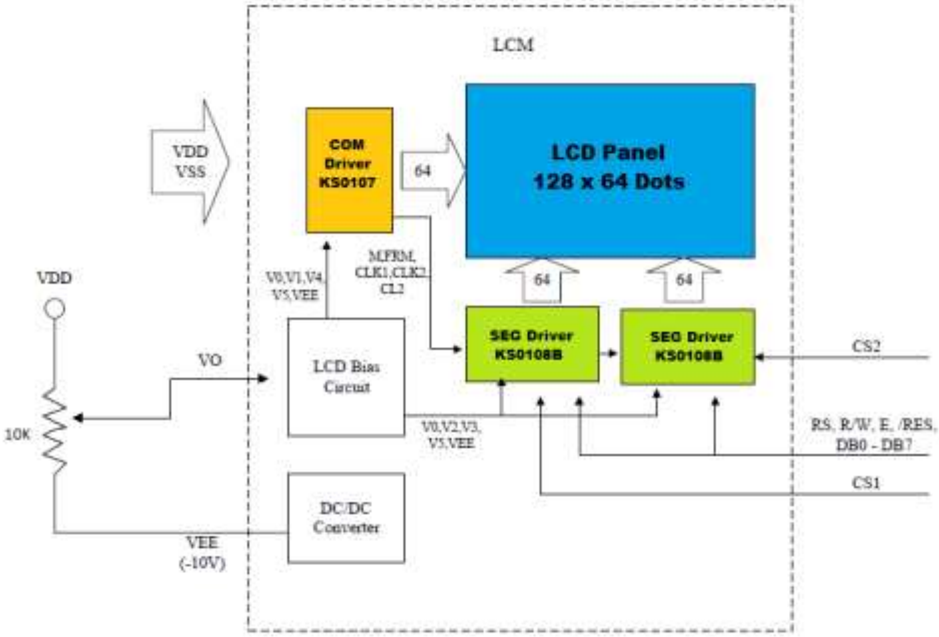
EN	P0.10	CN18.10	GLCD Enable Signal
R/W	P0.11	CN18.9	GLCD Read / Write
VCC		CN18.1, CN18.3, CN18.19	Power Supply
GND		CN18.2, CN18.4, CN18.20	Power Supply

Pin Diagram:



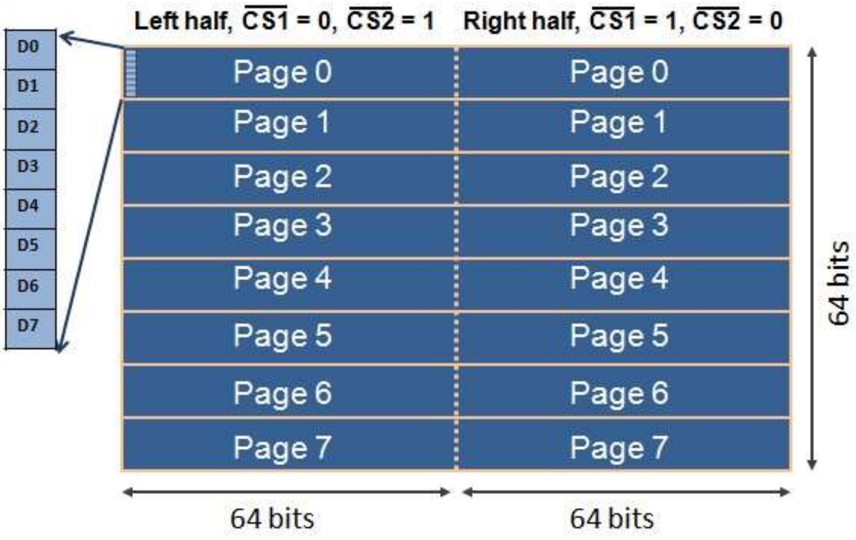
The 128X64 Graphical LCD interfaces to adjust contrast through trim pot. The GLCD needed to create 8-bit interface; 8 data bits (D0 – D7), three control lines, address bit (RS), read/write bit (R/W) and control signal (E), Page Select (CS).

The graphical LCD used isa 128X64 module, which is a 128×64 pixel monochromatic display. It uses KS0108B (or KS0107B) display controller. The KS0108B is a dot matrix LCD segment driver with 64 channel output, and therefore, the GLCD module contains two sets of it to drive 128 segments. On the other hand, the KS0107B is a 64-channel common driver which generates the timing signal to control the two KS0108B segment drivers. The KS0108B and KS0107B are very popular controllers and have made their way into many graphical LCDs. The internal block diagram of the AMPIRE 128X64 GLCD module is shown below.

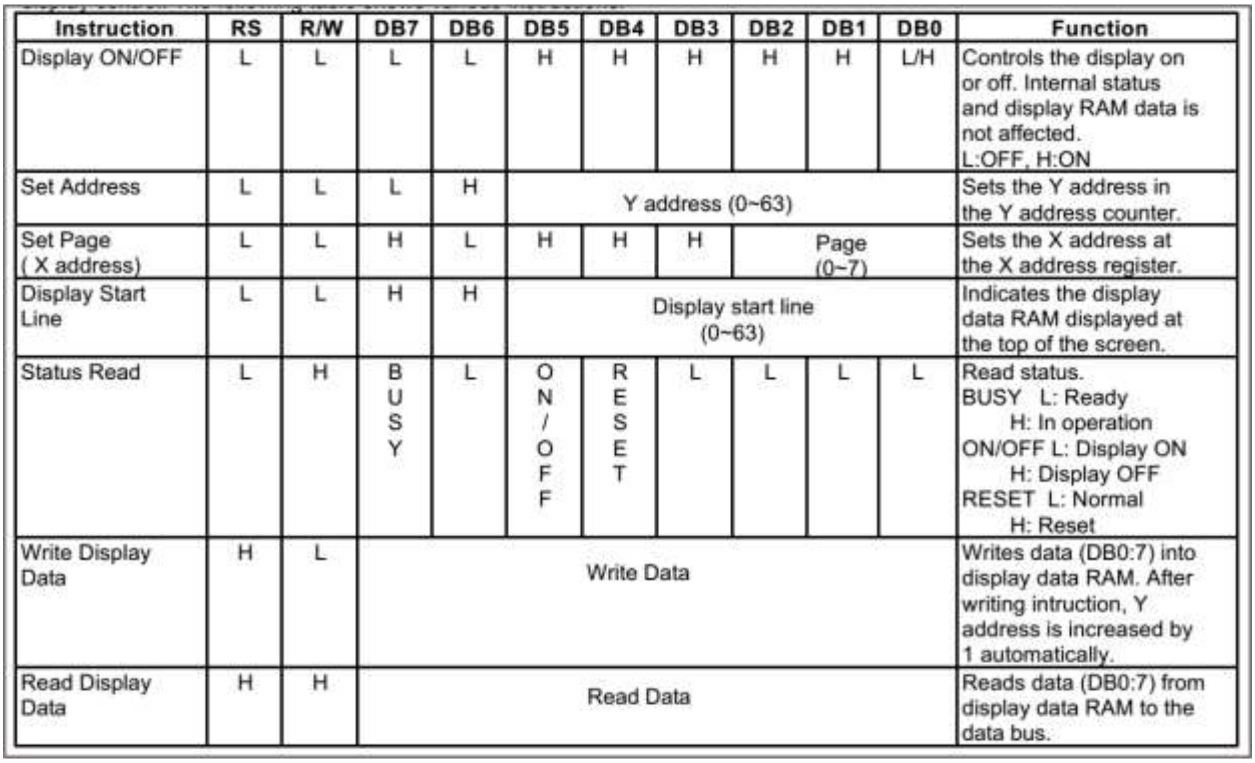


The KS0107 drives the 64 display lines, COM1 – COM64. The first KS0108 drives the left half segments (SEG1 to SEG64) and the second one drives the right half segments (SEG65 to SEG128) of the display. The two halves of the display can be individually accessed through the chip select pins (CS1 and CS2) of the two KS0108 drivers. Each half consists of 8 horizontal pages (0-7) which are 8 bits (1 byte) high. This is illustrated in the drawing below.

Page Structure of GLCD:



GLCD Instructions:



Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Function
Display ON/OFF	L	L	L	L	H	H	H	H	H	L/H	Controls the display on or off. Internal status and display RAM data is not affected. L:OFF, H:ON
Set Address	L	L	L	H	Y address (0~63)						Sets the Y address in the Y address counter.
Set Page (X address)	L	L	H	L	H	H	H	Page (0~7)			Sets the X address at the X address register.
Display Start Line	L	L	H	H	Display start line (0~63)						Indicates the display data RAM displayed at the top of the screen.
Status Read	L	H	BUSY	L	ON/OFF	RESET	L	L	L	L	Read status. BUSY L: Ready H: In operation ON/OFF L: Display ON H: Display OFF RESET L: Normal H: Reset
Write Display Data	H	L	Write Data								Writes data (DB0:7) into display data RAM. After writing intruction, Y address is increased by 1 automatically.
Read Display Data	H	H	Read Data								Reads data (DB0:7) from display data RAM to the data bus.

CONCLUSION:

QUESTIONS:

1) List applications of ARM processor.

2) Compare ARM7, ARM9 and ARM11. List the applications of these processors.

3) What is mean by 7 TDMI w.r.t ARM core.

EXPERIMENT NO. 5

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Using UART of LPC2148 for serial reception and transmission from/to computer.

OBJECTIVES:

- To understand serial transmission protocol UART.
- To interface UART based devices with LPC2148.
- To program LPC2148 for implementing desired application.

CO and PO MAPPED :

CO –

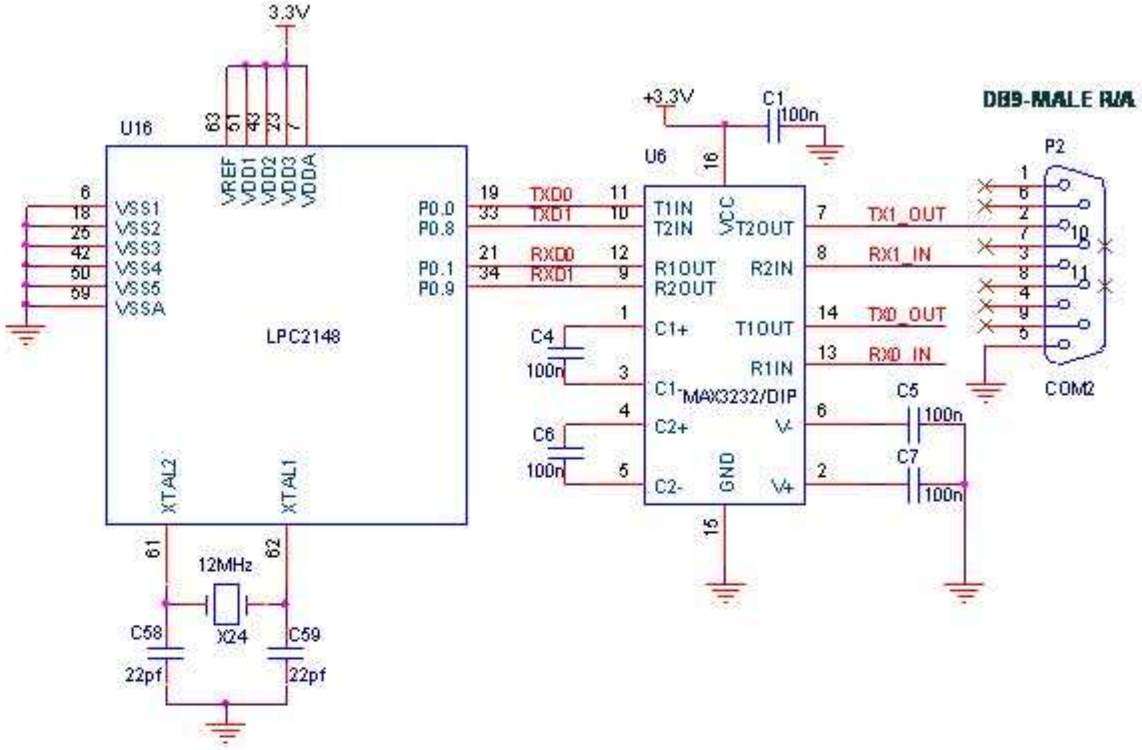
PO –

REQUIREMENT: LPC2148 board,12v adaptor,10 pin FRC, Serial Cable or USB cable.

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board. Note down the COM port assigned in device manager.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Connect the HyperTerminal or flash magic terminal to the COM port mentioned.
- Press reset on the kit.
- O/P display on Terminal.

INTERFACING DIAGRAM:



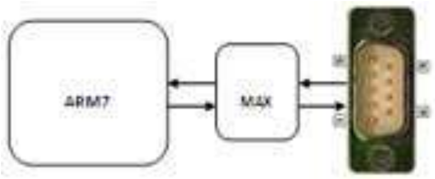
THEORY:

UART (Universal Asynchronous Receiver Transmitter) are one of the basic interfaces which provide a cost effective simple and reliable communication between one controller to another controller or between a controller and PC.

Usually all the digital ICs work on TTL or CMOS voltage levels which cannot be used to communicate over RS-232 protocol. So a voltage or level converter is needed which can convert TTL to RS232 and RS232 to TTL voltage levels. The most commonly used RS-232 level converter is MAX232. This IC includes charge pump which can generate RS232 voltage levels (-10V and +10V) from 5V power supply. It also includes two receiver and two transmitters and is capable of full-duplex UART/USART communication. RS-232 communication enables point-to-point data transfer. It is commonly used in data acquisition applications, for the transfer of data between the microcontroller and a PC. The voltage levels of a microcontroller and PC are not directly compatible with those of RS-232, a level transition buffer such as MAX232 be used.

LPC2148 ARM-7 Core is having two UART in it. Built in Features of UART0 in LPC2148:-

- 16 byte Receive and Transmit FIFOs
- Register locations conform to '550 industry standard.
- Receiver FIFO trigger points at 1, 4, 8, and 14 bytes.
- Built-in fractional baud rate generator with auto bauding capabilities.
- Mechanism that enables software and hardware flow control implementation.

	UART DB-9 Connector	**LPC2148 Processor Lines**	**Serial Port Section**
UART0(P1) **ISP PGM**	TXD-0	P0.0	
	RXD-0	P0.1	
UART1 **(P2)**	TXD-1	P0.8	
	RXD-1	P0.9	

UART Registers of LPC2148

Register	**Description**
UxRBR	Contains the recently received Data
UxTHR	Contains the data to be transmitted
UxFCR	FIFO Control Register
UxLCR	Controls the UART frame formatting(Number of Data Bits, Stop bits)
UxDLL	Least Significant Byte of the UART baud rate generator value.
UxDLM	Most Significant Byte of the UART baud rate generator value.

LPC2148 generates the baud rate depending on the values of DLM,DLL.

$$Baud\_rate = \frac{\text{PCLK}}{(16 * (256 * DLM + DLL) * (1 + DivAddVal/MulVal))}$$

Both UART0 & UART1 blocks internally have a 16-byte FIFO (First In First Out) structure to hold the Rx and Tx data. Each byte in this FIFO represents a character which was sent or received in order. Both blocks also contain 2 registers each, for data access and assembly.

Tx has THR(Transmit Holding Register) and TSR(Transmit Shift Register) – When we write Data to be sent into THR it is then transferred to TSR which assembles the data to be transmitted via Tx Pin.

Similarly Rx has RSR(Receive Shift Register) and RBR(Receive Buffer Register) – When a valid data is Received at Rx Pin it is first assembled in RSR and then passed in to Rx FIFO which can be then accessed via RBR.

Steps for Configuring UART0

- Configure the GPIO pin for UART0 function using PINSEL register.
- Configure the FCR for enabling the FIXO and Reste both the Rx/Tx FIFO.
- Configure LCR for 8-data bits, 1 Stop bit, Disable Parity and Enable DLAB.
- Calculate the DLM, DLL values for required baud-rate from PCLK.
- Update the DLM,DLL with the calculated values.
- Finally clear DLAB to disable the access to DLM,DLL.

After this, the UART will be ready to Transmit/Receive Data at the specified baud-rate.

CONCLUSION:

QUESTIONS:

1) List the features of UART0 in LPC2148.
2) Draw and explain the block diagram of UART.
3) Write down the difference between UART0 and UART1.

Pimpri Chinchwad Education Trust’s

Pimpri Chinchwad College of Engineering

EXPERIMENT NO. 6

Department: E&TC **Academic Year: 2023-24** **Semester: I**

Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Interfacing GSM with LPC2148 for sending and receiving message and voice call.

OBJECTIVES:

- To understand working of AT commands.
- To interface GSM device with LPC2148.
- To program LPC2148 for implementing desired application.

CO and PO MAPPED:

CO –

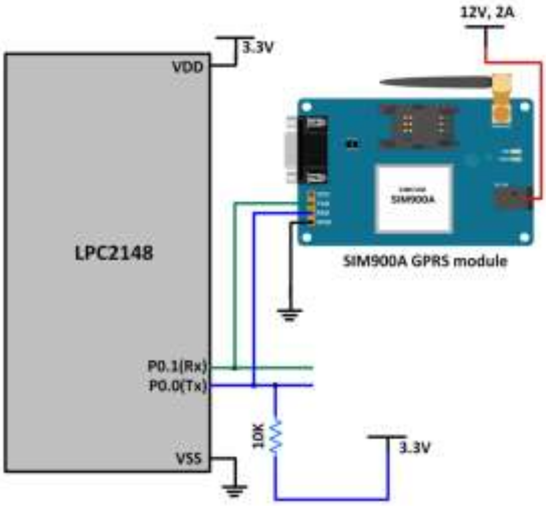
PO –

REQUIREMENT: LPC2148 board, 12v adaptor, GSM Modem, Serial Cable or USB cable.

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board. Note down the COM port assigned in device manager.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Connect the HyperTerminal or flash magic terminal to the COM port mentioned.
- Press reset on the kit.
- O/P display on Terminal.

INTERFACING DIAGRAM:



Micro controller’s RX (P0.1) is connected into GSM module’s TX and Micro controller’s TX (P0.0) is connected into GSM module’s RX pin. Some features of GSM MODEM include -

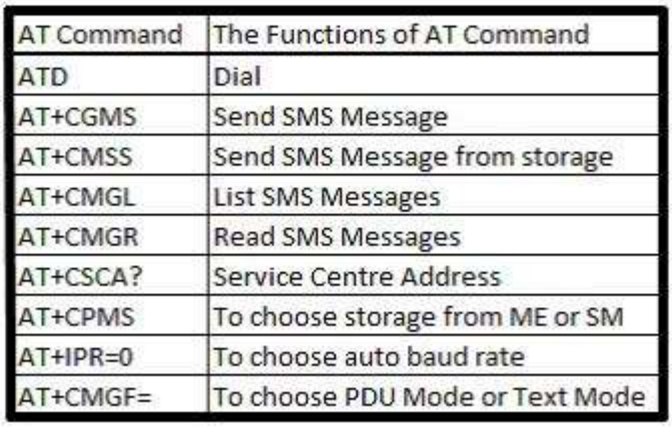
- Single supply voltage 3.2v-4.5v
- Typical power consumption in SLEEP Mode: 2.5mA.
- SIM300 tri-band
- MT,MO,CB, text and PDU mode, SMS storage: SIM card
- Supported SIM Card :1.8V,3V

THEORY:

GSM stands for Global System for Mobile Communication. It is a digital mobile telephony system. GSM digitizes and compresses data, then sends it down a channel with two other streams of user data, each in its own time slot. It operates at either the 900 MHz or 1800 MHz frequency band. There are various GSM modules available in the market like SIM900, SIM700, SIM800, SIM808, SIM5320 etc.

SIM900A module allows users to send/receive data over GPRS, send/receive SMS and make/receive voice calls.AT commands are instructions used to control a modem. AT is the abbreviation of ATtention. Every command line starts with "AT" or "at". Many of the commands that are used to control wired dial-up modems, such as ATD (Dial), ATA (Answer), ATH (Hook control) and ATO (Return to online data state), are also supported by GSM/GPRS modems and mobile phones. Besides this common AT command set, GSM/GPRS modems and mobile phones support an AT command set that is specific to the GSM technology, which includes SMS-related commands like AT+CMGS (Send SMS message), AT+CMSS (Send SMS message from storage), AT+CMGL (List SMS messages) and AT+CMGR (Read SMS messages).

GSM Commands



AT Command	The Functions of AT Command
ATD	Dial
AT+CGMS	Send SMS Message
AT+CMSS	Send SMS Message from storage
AT+CMGL	List SMS Messages
AT+CMGR	Read SMS Messages
AT+CSCA?	Service Centre Address
AT+CPMS	To choose storage from ME or SM
AT+IPR=0	To choose auto baud rate
AT+CMGF=	To choose PDU Mode or Text Mode

Given below are some of the tasks that can be done using AT commands with a GSM/GPRS modem or mobile phone:

1) Get basic information about the mobile phone or GSM/GPRS modem. For example, name of manufacturer (AT+CGMI), model number (AT+CGMM), IMEI number (International Mobile Equipment Identity) (AT+CGSN) and software version (AT+CGMR).
2) Get basic information about the subscriber. For example, MSISDN (AT+CNUM) and IMSI number (International Mobile Subscriber Identity) (AT+CIMI).
3) Get the current status of the mobile phone or GSM/GPRS modem. For example, mobile phone activity status (AT+CPAS), mobile network registration status (AT+CREG), radio signal strength (AT+CSQ), battery charge level and battery charging status (AT+CBC)
4) Establish a data connection or voice connection to a remote modem (ATD, ATA, etc).
5) Send and receive fax (ATD, ATA, AT+F*).
6) Send (AT+CMGS, AT+CMSS), read (AT+CMGR, AT+CMGL), write (AT+CMGW) or delete (AT+CMGD) SMS messages and obtain notifications of newly received SMS messages (AT+CNMI).
7) Read (AT+CPBR), write (AT+CPBW) or search (AT+CPBF) phonebook entries

Algorithm for GSM module interfacing with LPC2148

```
1) Start
2) Initialize UART0 or UART1 serial interface using followinginstruction
        PINSEL0=0X0000 0005;//Enable P0.0-TxD0,P0.1-RxD0
        U0LCR=0X83; //8-BIT Character length,NO parity,1 stop bit
        U0DLL=97; //Baud rate=9600@PCLK=15Mhz
        U0LCR=0X03;//Dlab=0
3) Transmit different AT commands through UART module using instruction
        while(!(U0LSR&0X20));//Monitor TI flag
4) If transmission buffer is Empty, Transmit AT commands
        U0THR=ch;
5) Provide delay while transmitting each command
6) To transmit a single character use PUTCH function & to transmit a string use PUTS
function
7) END
```

CONCLUSION:

QUESTIONS:

1) Explain interfacing of GSM using UART with LPC2148.
2) Explain how the baud rate is generated in UART0 of LPC2148.
3) Explain the working of Zigbee modem.
4) Explain interfacing of GPS using UART with LPC2148.

Pimpri Chinchwad Education Trust's

Pimpri Chinchwad College of Engineering

EXPERIMENT NO. 7

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Using built-in ADC of LPC2148 for displaying its values

OBJECTIVES:

- To understand LPC2148 microcontroller & its various ports and registers.
- To understand Analog to Digital Conversion (ADC).
- To interface Analog sensors like potentiometer, temperature sensor with LPC2148.
- To program LPC2148 for implementing desired application.

CO and PO MAPPED:

CO –

PO –

REQUIREMENT: LPC2148 board, 12v adaptor, 10 pin FRC, USB cable, Keil uVision4, Flash Magic

PROCEDURE: -

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Turn potentiometer P1 to show the result of ADC on the LCD.

INTERFACING DIAGRAM:

THEORY:

LPC2148 contains 2 analog to digital convertors, AD0 and AD1 with 6 and 8 channels respectively, i.e., there are total 14 analog inputs. The features of ADC are:

- 10 bit successive approximation analog to digital converter.
- Input multiplexing among 6 or 8 pins (ADC0 and ADC1).
- Power-down mode.
- Measurements range 0 V to V_{REF} (typically 3V; not to exceed V_{DDA}).
- 10 bit conversion time ≥ 2.44 µs.
- Burst conversion mode for single or multiple inputs.
- Optional conversion on transition on input pin or Timer Match signal.
- Global Start command for both converters.

Basic clocking for the A/D converters is provided by the VPB clock. A programmable divider is included in each converter, to scale this clock to the 4.5 MHz (max) clock needed by the successive approximation process. A fully accurate conversion requires 11 of these clocks.

AD0.1 ~ AD0.7
AD1.0 ~ AD1.7 - These are the analog input pins. While the ADC pins are specified as 5 V tolerant in the datasheet, more than 3.3 V (V_{DDA}) +10 % should not be applied to any pin that is selected as an ADC input.

VREF - This pin provides a reference voltage for A/C conversion.

VDDA, VSSA - Same as V_{DD} and V_{SS} but should be isolated to minimize noise and error.

ADC registers:

ADCR	A/D Control Register. The ADCR register must be written to select the operating mode before A/D conversion can occur. ADC channel, 4.5MHz A/D clock, burst mode, and number of result bits and conversion start conditions are set in this register.
ADGDR	A/D Global Data Register. This register contains the ADC's DONE bit and the result and channel number of the most recent A/D conversion.
ADSTAT	A/D Status Register. This register contains DONE and OVERRUN flags for all of the A/D channels, as well as the A/D interrupt flag.
ADINTEN	A/D Interrupt Enable Register. This register contains enable bits that allow the DONE flag of each A/D channel to be included or excluded from contributing to the generation of an A/D interrupt.
ADDR0 – ADDR7	A/D Channel 0~7 Data Register. This register contains the result and DONE bit of the most recent conversion completed on channel 0~7.

Algorithm :

1. Initialize ADC.
2. Initialize timer.
3. Initialize interrupt.
4. Start timer.
5. If timer match interrupt occurs, start ADC conversion and display result.
6. Reset timer.
7. Go to step 5.

CONCLUSION:

QUESTIONS:

1. Explain the necessity of Vectored Interrupt Controller along with the working of VIC in LPC2148 along with SFRs
2. Explain the ADCR in detail. What value to be loaded in ADCR for following specification

   ADC port pin : AD0.1 i.e. Channel 1 of ADC0

   Desired ADC clock = 3 MHz

   Desired Precision = 11 bits
3. Write an embedded C program to interface analog voltage to channel 7 of inbuilt ADC1 of LPC2148. The Pclk is 12MHz

EXPERIMENT NO. 8

Department: E&TC **Academic Year: 2023-24** **Semester: I**
Year: TY (A,B,C) **Subject: Advanced Microcontroller**

TITLE:

Waveform Generation using DAC

OBJECTIVES:

- To understand the use of Digital to Analog Converter.
- To understand interfacing of analog sensors with LPC2148 microcontrollers.

CO and PO MAPPED :

CO –

PO –

REQUIREMENT: LPC2148 board, 12v adaptor, 10 pin FRC, USB cable, Keil uVision4, Flash Magic

PROCEDURE:-

- Connect the 12V adaptor to LPC 2148 and the board.
- Connect the serial or USB cable to LPC 2148 board.
- Open the Flash Magic software and make setting as per given.
- Make sure that programming switch on LPC 2148 is set to RUN mode.
- Press reset on the kit.
- Download the Hex file to LPC 2148 Board.
- Observe the waveform on DSO.

INTERFACING DIAGRAM:

THEORY:

The digital to analog converter is a device widely used to convert digital pulses to analog signals. The two methods of creating DAC are binary weighted and R -2R ladder. DAC 0808 uses the R-2R method since it can achieve a high degree of precision. The first criterion for judging a DAC is its resolution, which is the function of the number of binary inputs. The common ones are 8, 10 and 12 bits. The number of data bit inputs decides the resolution of the DAC since the number of analog output levels is equal to 2n, where n is the number of data inputs. DAC 0808 provides 256 discrete voltage or current levels of output. In DAC 0808, the digital inputs are converted into current Iout and by connecting a resistor to Ioutpin, we convert the result to voltage. The total current provided by IOUT pin is a function of binary numbers at the D0-D7 pins inputs to DAC 0808 and reference current (Iref) is as follows:

Iout=Iref (D7/2+D6/4+D5/8+D4/16+D3/32...+D0/256)

Where D0 is the LSB, D7 is the MSB for the inputs and Iref is the input current that must be applied.

Microcontroller are used in wide variety of applications like for measuring and control of physical quantity like temperature, pressure, speed, distance, etc.In these systems microcontroller generates output which is in digital form but the controlling system requires analog signal as they don't accept digital data thus making it necessary to use DAC which converts digital data into equivalent analog voltage.The use 8-bit DAC 0808. This IC converts digital data into equivalent analog Current. Hence we require an I to V converter to convert this current into equivalent voltage.

ALGORITHM FOR WAVEFORM GENERATION USING DAC OF LPC2148:

Square Waveform Algorithm:

Step1: Select Port 0 pin (P0.25) to DAC output using PINSEL1 register.
Step2: Load the digital value with 0x00H or 0d.
Step3: Send the data to the DACR to get the output.
Step4: See the result on P0.25 pin and observe the waveforms in a CRO.
Step5: Call delay.
Step6: Load the digital value with 0x00H or 0d.
Step7: Call delay.
Step8: Repeat step 2 to 8.
Step9: Repeat step 3 to 8.

Triangular Waveform Algorithm:

Step1: Set the DAC output pin as Port 0 (P0.25) pin by PINSEL1 register.
Step2: To generate Triangular wave take values in a for loop from 0 and end with 1023 in incrementing order. Now take another loop from 1023 to 0 in decrementing order. This will give the triangular wave.
Step3: Send the data to the DACR to get the output.
Step4: See the result on P0.25 pin and observe the waveforms in a CRO.

Staircase Waveform Algorithm:

Step1: Divide the max value (255) by the no of steps required to generate a staircase wave
Step2: Clear the accumulator.
Step3: Transmit the value to the port.
Step4: Increment the value of accumulator by the no obtained after division in step 1.
Step5: Compare the value with 255 and go to step 3 if the value is less than 255 else go to step 2

CONCLUSION:

QUESTIONS:

1. Write short note on : On chip DAC for LPC2148.
2. Explain the two methods for creating DAC output.
3. Write a program to generate a staircase waveform.